

Uniconf v.1.0.0

Copyright: Yury Minenkov
N.N. Semenov Federal Research Center
of Chemical Physics RAS
119991 Moscow, The Russian Federation
Kosygina street, 4

E-mail: Yury.Minenkov@chph.ras.ru, Yury.Minenkov@gmail.com

Table of Contents

1. Brief theoretical introduction.....	2
2. Quick start.....	3
2.1 Prerequisites.....	3
2. Main Keywords.....	3
2.1 Input file general design.....	3
2.2 Group \$BOND.....	3
2.3 Group \$MOLECULE.....	4
Cartesian coordinates of all atoms should be provided in this section.....	4
2.4 Group \$CHARGE.....	4
Atomic charges for each atom in \$MOLECULE group should be provided here.....	4
2.2 Group \$ALGO.....	4
2. Detailed documentation (advanced keywords).....	7

1. Brief theoretical introduction

Uniconf generator algorithm

Input: Starting cluster geometry $\{X,Y,Z\}_0$, atomic charges $\{Z_i\}$, rotatable bonds N_{ROT}

Output: N_{CONF} cluster geometries $\{X,Y,Z\}_i$

// determination of connectivity in the molecule

Connectivity: write (automatic) or read

Initialization:

for i in range(N_{CONF}) **do:**

 // setting up random values

$6N_{\text{FRG}} + N_{\text{ROT}} \leftarrow \text{rand}()$

Procedure: preliminary generation

for i in range(N_{CONF}) **do:**

 // optimization with N_{ITER1} steps

$\{X,Y,Z\}_i \leftarrow \min(6N_{\text{FRG}} + N_{\text{ROT}}, N_{\text{ITER1}})$

 // shortest non-bonded distance check

if mindist ($\{X,Y,Z\}_i$) < thld:

remove ($\{X,Y,Z\}_i$)

else:

pass

Clustering ($\{X,Y,Z\}_i$, k_1)

$N_{\text{CONF}} = k_1$

End Procedure

Procedure: final generation

for i in range(N_{CONF}) **do:**

 // optimization with N_{ITER2} steps

$\{X,Y,Z\}_i \leftarrow \min(6N_{\text{FRG}} + N_{\text{ROT}}, N_{\text{ITER2}})$

for i in range(N_{CONF}) **do:**

for j in range($N_{\text{CONF}}-1$) **do:**

 // close in energies?

if $|E(\{X,Y,Z\}_i) - E(\{X,Y,Z\}_j)| < \text{thld}$:

if RSMD($\{X,Y,Z\}_i - \{X,Y,Z\}_j$) < thld:

$N_{\text{CONF}} = N_{\text{CONF}} - 1$

remove ($\{X,Y,Z\}_j$)

Clustering ($\{X,Y,Z\}_i$, k_2)

$N_{\text{CONF}} = k_2$

End Procedure

return $N_{\text{CONF}} \{X,Y,Z\}_i$

Scheme 1. Pseudo code of the *uniconf* spatial structure generator.

2. Quick start

2.1 Prerequisites

- 64bit Linux OS
- Conformer generator binary (uniconf) & input file (filename.inp)
- Input file should contain:
 - ✗ Rotatable bonds (if any)
 - ✗ Initial Cartesian coordinates of the (super-)molecule
 - ✗ Atomic charges
- Running the code: “/path/to/program/uniconf filename.inp”, e.g.
~/Isomers/128/uniconf m1_01_Li_VC_N3.inp

2. Main Keywords

2.1 Input file general design

Each keyword group starts with “\$” symbol and should be closed with “\$END” expression.

```
$BOND
28 29 120 240 15
..
$END
$ALGO
..
$END
$MOLECULE
Li   -3.118257632    1.717973895    0.890951148
C    0.543523281    0.285069258   -0.272235148
..
$END
$CHARGE
1.0
0.778632
..
$END
```

2.2 Group \$BOND

Here the rotational bonds should be specified in the following format:

Atom1	Atom2	Angle_Increment	Angle_MAX_value	Angle_MAX_deviation
1	2	120	240	15

In the example above the rotational bond is between atom 1 and atom 2. This torsion is specified to have three values: 0, 120, 240. With 0 value here the starting torsion angle value is meant.

In case of no rotatable bonds, leave this group empty.

2.3 Group \$MOLECULE

Cartesian coordinates of all atoms should be provided in this section.

2.4 Group \$CHARGE

Atomic charges for each atom in \$MOLECULE group should be provided here.

2.2 Group \$ALGO

```
CONNECT  =WRITE      // automatic connectivity detection that is written to filename.bond
                        // filename.bond can be modified to adjust/correct the connectivity
                        // (default)
                        =READ      // connectivity is read from filename.bond
                        // needed in cases when automatic connectivity fails

ALGO      =CONF       // generate conformers (default)

MK         =int        // integer number (e.g. 10) N of initially generated conformers on-the-fly;
                        // initial NCONF value in the pseudo code above
                        =0        // systematic conformational search with respect to provided torsions
                        // see $BOND group above

COM        =int        // what is considered to be center of coordinates
                        // -1 center of mass of the whole cluster geometry specified in
                        // $MOLECULE (default)
                        // 0 .. N – center of mass of individual fragment in the cluster
                        // numeration starts from 0. It is convenient when an ion needs to
                        // be surrounded by molecules, e.g. Li+

SCALE      =float      // any float value from 0.0 to 0.99 (default: 0.75)
                        // characterizes the maximum of individual fragment com from
                        // the initial center. The lesser the value the closer to the initial
                        // structure conformer is generated.

LIBRARY    =NLOPT      // optimization library for potential energy optimization (default)
                        // more at https://nlopt.readthedocs.io/en/latest/
                        OPTGLOB   =none      // no global optimization (recommended) (default)
                        =nlopt_gn_direct_l
                        =nlopt_gn_direct_l_rand
                        =nlopt_gn_direct_noscal
                        =nlopt_gn_direct_l_rand_noscal
                        =nlopt_gn_orig_direct
                        =nlopt_gn_orig_direct_l
                        =nlopt_gn_crs2_lm
                        =nlopt_gn_msl
                        =nlopt_gn_msl_lds
                        =nlopt_gn_isres
                        =nlopt_auglag
```

		=nlopt_auglag_eq =nlopt_gn_esch OPTLOC =nlopt_ln_sbplx <i>// local optimization method, must be specified! (default)</i> =nlopt_ln_praxis =nlopt_ln_cobyla =nlopt_ln_newuoa =nlopt_ln_newuoa_bound =nlopt_ln_neldermead =nlopt_ln_sbplx =nlopt_ln_auglag =nlopt_ln_auglag_eq =nlopt_ln_bobyqa
ITER1	=int	<i>// number of iterations for the first (preliminary) conformer generation // (default: 10)</i>
DIST	=float	<i>// minimal non-bonded distance threshold for conformer exclusion // after preliminary and final conformer generation (default: 1.25)</i>
NONHD	=FALSE	<i>// DIST related keyword. When searching for the minimal non-bonded // distance, do not skip the pairs including hydrogens (e.g. C – H) // (default)</i>
	=TRUE	<i>// skip the pairs that contain hydrogens</i>
SORT	=e	<i>// after minimal distance test (see DIST) that occurs after preliminary // and final conformer generation, all conformers are sorted based on // certain property. E stands for sorting by energies (default)</i>
	=d1	<i>// sort by the average distance from each fragment's com to the whole // molecule center of mass</i>
	=d2	<i>// same as d1 but in case of COM!=-1 (i.e. when center is not the whole // molecule center of mass)</i>
MAXCONF	=int	<i>// after sorting above, take that amount of conformer is taken // for further treatment (default: 100)</i>
KMEANS	=int	<i>// first clustering option (in the preliminary conformer generation) // -3: K (number of clusters) is estimated $K=Nconf/2$ // -1: (default) no clustering // 0: $K = (Nconf/2)**0.5$ // Any int number, e.g. 10</i>
KMEANS1	=int	<i>// second clustering option (after the final conformer generation) // only for STRATEGY=1, see below // -3: K (number of clusters) is estimated $K=Nconf/2$ // -1: (default) no clustering // 0: $K = (Nconf/2)**0.5$ // Any int number, e.g. 10</i>
CLUSTERPROPERTY1	=IXX	<i>// the clustering property; the whole molecule sorted principal // moments of inertia (3)</i>
	=distcommol	<i>// distance from each individual fragment com to the // molecular COM</i>
	=ixxfrg	<i>// three sorted principal moments of the whole molecules</i>

		<i>// and each individual fragment (default)</i>
	=eixx	<i>// the whole molecule energy and three principal moments // of inertia</i>
	=trans	<i>// translational variables for each fragment</i>
	=rot	<i>// rotational variables for each fragment</i>
	=transrotbond	<i>// translational, rotational variables and torsion bonds // for each fragments</i>
ITER2	=int	<i>// number of iterations for the second (final) conformer generation // (default: 10 000)</i>
EDIFF	=float	<i>// energy difference floating point value below which the // structure geometries are compared with quatit RMSD // (default: 0)</i>
RMSD	=float	<i>// RMSD criteria for difference between the two structures for discarding // at the DISCARDING procedure after final conformer generation // utilized by Quatfit algorithm</i>
MAXCONFPRINT	=int	<i>// number of conformers for final printing to the file</i>
STRATEGY	=0	<i>// strategies to treat the conformers after the final generation (default: 0) // just print MAXCONFPRINT conformers to the file</i>
	=1	<i>// clustering of all conformers after the final generation, followed by // sorting. After that MAXCONFPRINT conformers are printed to file.</i>

2. Detailed documentation (advanced keywords)

ALGO	=OPTIMIZE	// simple optimization of a given structures // no discrete values of bond torsions are allowed
ITER2	=int	// the number of geometry optimization steps for ALGO=OPTIMIZE // is given with ITER2
PAREN	=FALSE	// Parallel mode for potential energy single-point calculation (default: // FALSE)
	=TRUE	// DO NOT USE THIS, NO SPEED UP IS OBSERVED
ITER	=int	// clustering-related iteration number (default: 1)
TESTSIZE	=float	// NLOPT optimization-related keyword (default: 1e-3)
KEL	=float	// scaling coefficient for electrostatic energy (default: 1)
KGD	=float	// scaling coefficient for the DFT-D2 Grimme's dispersion // sometimes it helps for stronger binding of the fragments (default 0)
KVW	=float	// scaling coefficient for the van der Waals UFF energy (default: 1)
HB1	=float	// A hydrogen bonding coefficient, see main article (default: 6.71)
HB2	=float	// B hydrogen bonding coefficient, see main article (default: 3.55)
HB3	=float	// alpha hydrogen bonding coefficient, see main article (default: 0.75)
ALGO	=FIT	// Fitting parameters to reproduce the initial geometry specified in // \$MOLECULE group
FIT	=false	// no fitting (default)
	=KEL	// fitting electrostatic energy coefficient
	=KVW	// fitting van der Waals non-bonded UFF energy
	=KELKVW	// fitting both electrostatic and van der Waals energies
	=HBOND	// fitting the hydrogen bond coefficients
ITERFIT	=int	// number of iterations for fitting (default: 1000)
ITERFIRGEOM	=int	// number of geometry optimization steps inside fitting step // (default 1000)
OPTGLOBFIT	=str	// global optimization algorithm for fitting (default: // nlopt_gn_orig_direct)
OPTLOCFIT	=str	// local optimization algorithm for fitting (default: // nlopt_ln_sbplx)
OPTLOCFITGEOM	=str	// local optimization algorithm for geometry optimization // inside fitting (default: nlopt_ln_sbplx)
CLUSTERALGORITHM	=K-MEANS	// (default) algorithms for clustering
	=K-MEDOIDS	// (memory intensive, use with caution)
PRINTDATA	=0	// (default) no detailed output for clusterization
	=1	// detailed output for clusterization
CENTROIDCENTER	=0	// (default) do not use the centroid center, use minimum E // structure
	=1	// use the cluster centroid
PRINTLEVEL	=0	// (default: 0). Do not print coordinates after ITER1 & ITER2

```
=-1 // print all coordinates at the intermediate steps.
```